

Universidad de San Carlos de Guatemala  
Facultad de Ingeniería  
Escuela de Ciencias y Sistemas  
Análisis y Diseño de Sistemas 1

**PRÁCTICA 1 MANUAL TÉCNICO**  
**GRUPO 3**

Juan Antonio Solares Samayoa	201800496
Josue Guillermo Orellana Cifuentes	201801366
Carlos Ojani Ng Valladares	201801434
Carlos Agustín Ché Mijangos	201800624
Gerson Alejandro Beltetón Urbina	201807228

## DETALLES DE LA APLICACIÓN Y TECNOLOGÍAS UTILIZADAS

---

Se realizó un servidor en Nodejs para realizar las siguientes funciones: fibonacci, determinar si un número es primo, determinar si una palabra es palíndromo, raíz cúbica, potencia al cubo, multiplicación y división, para ello se inició a partir de una rama Main de la cual se creó una rama developer, a partir de esta se crearon distintas ramas feature por cada función que se mencionó anteriormente, una vez creadas las funciones en las ramas feature se hizo merge a developer por cada feature, por cada cambio se hizo una rama release y posteriormente se hizo merge a la rama Main.

Se realizaron ramas bugfix y hotfix para corregir errores y agregar cambios, en el caso del hotfix se utilizó para corregir la funcionalidad de la división entre 0, para ello se creó una rama a partir de Main y posteriormente se hizo merge a developer y a main. Los bugfix se crearon para agregar una funcionalidad a cada función, en este caso se agregó un mensaje en la respuesta de cada api, para el bugfix se creó una rama a partir de developer y una vez agregada la funcionalidad se realizó un merge a developer.

- Para el backend se utilizó Nodejs
- Para el frontend se utilizó Angular
- Para el sistema de versionamiento se utilizó Github y Gitkraken

Se utilizó Nodejs del lado del backend por cuestiones demostrativas. También por su fácil manejo y escalabilidad al momento de generar aplicaciones, ya que al utilizar el framework de Express en conjunto con Nodejs, se incrementa la velocidad de desarrollo y se aprovecha una de sus grandes ventajas que es el manejo de rutas.

Para el frontend, Angular. Al ser un framework basado en una distribución modular, facilita el desarrollo, manejo e identificación de errores. Por último, se utilizó Github para el host de nuestro repositorio por la experiencia que se tiene utilizándolo entre los miembros del grupo. Así como Gitkraken para manejar el repositorio, este nos permite manejar y visualizar de manera sencilla y óptima el flujo de trabajo de la aplicación.

The diagram illustrates a structured Git branching strategy. It begins with a 'first commit' on the 'main' branch. Subsequent releases (e.g., 1.0.0, 1.1.0, 1.2.0) are created by merging the 'develop' branch back into 'main'. New features are developed on branches prefixed with 'f/' (e.g., f/1.0.0-201801366). Bugfixes are implemented by creating a branch from 'main', addressing the issue, and then merging the changes back into both 'main' and 'develop'. The diagram also shows the use of 'hotfixes' for specific versions (e.g., 2.0.0/hotfix\_division) and the process of merging changes from 'develop' back into 'main' for production-ready code.

# Aplicación Web

La aplicación web consume los endpoints desarrollados anteriormente, cuenta con un selector de opciones, por lo que cada opción representa una funcionalidad de manera individual.

## Versionamiento

Feature Palindromo	▼
Feature Fibonacci	▼
Feature Primos	▼
Feature Potencia	▼
Feature Raiz	▼
Feature Multiplicación	▼
Feature División	▼

## Versionamiento

Feature Palindromo	^
<b>Feature Palindromo.</b> Función que evalua si una palabra es palindroma o no.	
Ingrese la palabra a evaluar	
<input type="text" value="Palabra"/>	
<button>Validar Palindromo</button>	
Feature Fibonacci	▼
Feature Primos	▼
Feature Potencia	▼